Probabilistic Latent Tensor Factorization

Y. Kenan Yılmaz
kenan@sibnet.com.tr

A. Taylan Cemgil
taylan.cemgil@boun.edu.tr

Department of Computer Engineering, Boğaziçi University, 34342, Istanbul, Turkey

Abstract

We develop a probabilistic modeling framework for multiway arrays. Our framework exploits the link between graphical models and tensor factorization models and it can realize any arbitrary tensor factorization structure, besides many popular models such as CP or TUCKER models with Euclidean error and for nonnegativity with KL error. The probabilistic framework enables us to develop a model selection methodology based on variational Bayes.

1 The Model : Probabilistic Latent Tensor Factorization (PLTF)

We propose a unifying framework for full Bayesian inference in which any arbitrary tensor factorization structure for Euclidean and KL costs can be realized. By making use of the duality between exponential families and Bregman divergences, here we cast the *Tensor Factorization* problem as inference problem in the probabilistic *Graphical Models* with Gaussian or Poisson components. This way the tensor factorisation reduces to a parameter estimation problem. The associated inference algorithms can be derived automatically via message passing using matrix computation primitives and the model can be selected via a variational bound on the marginal likelihood [2, 1]. For this purpose, we introduce a notation for tensor factorization models that closely resembles undirected probabilistic graphical models [2, 9].

Probabilistic Latent Tensor Factorization is introduced first, in the LVA/ICA Conference, held in France on 27-30th September 2010 [10]. Apart from that introduction, for NIPS 2010 TF workshop we include model priors as hyperparameters as well as model order determination and model selection as a natural extension to PLTF.

1.1 Notation

Following the established jargon, we call a K-way array $X \in \mathcal{X}^{I_1 \times I_2 \times \cdots \times I_K}$ simply a 'tensor'. Here, I_k are finite index sets, where i_k is the corresponding index. We denote an element of the tensor $X(i_1, i_2, \ldots, i_K) \in \mathcal{X}$ as $X^{i_1,i_2,\ldots,i_K}$. Similarly, given the index set $W = \{i_1, \ldots, i_K\}$ we use the notation $X(w)$ to denote an element of $X^{i_1,i_2,\ldots,i_K}$. We associate with each TF model an undirected graph, where each vertex corresponds to an index. We let V be the set of vertices $V = \{v_1, \ldots v_\alpha, \ldots, v_N\}$. Our objective is to estimate a set of tensors $\mathcal{Z} = \{Z_\alpha | \alpha = 1 \ldots N\}$ such that

$$\text{minimize } D(X||\hat{X}) \text{ s.t. } \hat{X}(w) = \sum_{\bar{w} \in \bar{W}} \prod_{\alpha} Z_\alpha(v_\alpha) \tag{1}$$

where the function D is a divergence. Each Z_α is associated with an index set V_α such that $V = \cup_\alpha V_\alpha$. Two distinct sets V_α and $V_{\alpha'}$ can have nonempty intersection but they don't contain each other. We define a set of 'visible' indices $W \subseteq V$ and 'invisible' indices $\bar{W} \subseteq V$ such that $W \cup \bar{W} = V$ and $W \cap \bar{W} = \emptyset$.

Example 1 *For $V = \{i, j, k, p, q, r\}$, $W = \{i, j, k\}$, $V_1 = \{i, p\}$, $V_2 = \{j, q\}$, $V_3 = \{k, r\}$ and $V_4 = \{p, q, r\}$ (core tensor) TUCKER3 model is*

$$\hat{X}^{i,j,k} = \sum_{p,q,r} Z_1^{i,p} Z_2^{j,q} Z_3^{k,r} Z_4^{p,q,r} \tag{2}$$

For $PLTF$, we write the following generative model.

$$\Lambda(v) = \prod_{\alpha}^{N} Z_\alpha(v_\alpha) \qquad \text{model paramaters to estimate} \tag{3}$$

$$S(w, \bar{w}) \sim \mathcal{PO}(S; \Lambda(v)) \qquad \text{element of latent tensor for } PLTF_{KL} \tag{4}$$

$$S(w, \bar{w}) \sim \mathcal{N}(S; \Lambda(v), 1) \qquad \text{element of latent tensor for } PLTF_{EU} \tag{5}$$

$$X(w) = \sum_{\bar{w} \in \bar{W}} S(w, \bar{w}) \tag{6}$$

$$M(w) = \begin{cases} 0 & X(w) \text{ is missing} \\ 1 & \text{otherwise} \end{cases} \qquad \text{mask array} \tag{7}$$

Here, $\mathcal{N}()$ and $\mathcal{PO}()$ denote the Gaussian and the Poisson distributions. Λ is the intensity field and S is the latent source using Λ as the parameter and X is augmented from the source. Here note that $W \cup \bar{W} = \cup_\alpha V_\alpha = V$ and for their instantiations $(w, \bar{w}) = \cup_\alpha v_\alpha = v$. The Poisson distribution implies non negativity and also the model can deal naturally with the missing data [8, 3] .

1.2 $PLTF_{KL}$: Factorization Model for KL Error

Given this model, we optimize the log marginal likelihood $p(X|\Theta_{1:N})$ for Z_α using the EM that leads to the following fixed point update equation for Z_α

$$Z_\alpha(v_\alpha) \leftarrow Z_\alpha(v_\alpha) \frac{\sum_{v \notin V_\alpha} M(w) \frac{X(w)}{\hat{X}(w)} \prod_{\alpha' \neq \alpha} Z_{\alpha'}(v_{\alpha'})}{\sum_{v \notin V_\alpha} M(w) \prod_{\alpha' \neq \alpha} Z_{\alpha'}(v_{\alpha'})} \tag{8}$$

where $\hat{X}(w)$ is the model estimate given as

$$\hat{X}(w) = \sum_{\bar{w} \in \bar{W}} \prod_{\alpha} Z_\alpha(v_\alpha) \tag{9}$$

Example 2 (CP update equation) *The multiplicative update rule for CP (shortly for CANDECOMP/PARAFAC) [6] is generated by $PLTF_{KL}$ with the setting $N = 3$, $V = \{i, j, k, r\}$, $W = \{i, j, k\}$, $V_1 = \{i, r\}$, $V_2 = \{j, r\}$ and $V_3 = \{k, r\}$. The fixed point equation for Z_1 is*

$$Z_1^{i,r} \quad \leftarrow \quad Z_1^{i,r} \frac{\sum_{j,k} (M^{i,j,k} X^{i,j,k} / \hat{X}^{i,j,k}) Z_2^{j,r} Z_3^{k,r}}{\sum_{j,k} M^{i,j,k} Z_2^{j,r} Z_3^{k,r}} \tag{10}$$

1.3 Model Priors

After taking account the gamma model priors the fixed point update equation turns in to the following. One obvious use of the priors is that we can control the sparseness of the model.

$$Z_\alpha(v_\alpha) \sim \mathcal{G}(Z_\alpha; A_\alpha(v_\alpha), B_\alpha(v_\alpha)/A_\alpha(v_\alpha)) \tag{11}$$

$$Z_\alpha(v_\alpha) \leftarrow \frac{(A_\alpha(v_\alpha) - 1) + Z_\alpha(v_\alpha) \sum_{v \notin V_\alpha} M(w) \frac{X(w)}{\hat{X}(w)} \prod_{\alpha' \neq \alpha} Z_{\alpha'}(v_{\alpha'})}{\frac{A_\alpha(v_\alpha)}{B_\alpha(v_\alpha)} + \sum_{v \notin V_\alpha} M(w) \prod_{\alpha' \neq \alpha} Z_{\alpha'}(v_{\alpha'})} \tag{12}$$

1.4 Message Passing

We note that the update equation consists of structurally similar terms in both the denominator and the numerator. To exploit this we define the following the tensor valued function as $\Delta_\alpha(X)$:

$\mathbb{R}^{|X|} \rightarrow \mathbb{R}^{|Z_\alpha|}$

$$\Delta_\alpha(X) \equiv \left[\sum_{v \notin V_\alpha} \left(X(w) \prod_{\alpha' \neq \alpha} Z_{\alpha'}(v_{\alpha'}) \right) \right] \tag{13}$$

$\Delta_\alpha(X)$ is an object the same size of Z_α while $\Delta_\alpha(X)(v_\alpha)$ refers to a particular element of $\Delta_\alpha(X)$.

There are two benefits of using the Δ function notation. First, algebraically the Δ is equivalent to the computation of marginal potentials for the cliques in the Graphical Models, hence to compute Δ function one can use methods such as *variable elimination* or *junction tree*. The second benefit is that Δ function allows as to write element-wise fixed point update rule with tensor as compact as follows

$$Z_\alpha \leftarrow Z_\alpha \circ \frac{\Delta_\alpha(M \circ X / \hat{X})}{\Delta_\alpha(M)} \tag{14}$$

where $\circ$ and $/$ stand for element wise multiplication and division respectively. We also see that this equation can even turn into a matrix form and to be able to solve ALS equations.

1.5 $PLTF_{EU}$: Factorization Model for Euclidean Error

For Euclidean error the derivation of fixed point update equation is similar to that of KL divergence that we merely replace the Poisson likelihood with that of a Gaussian

$$\begin{aligned} \frac{\partial \mathcal{L}_{EU}}{\partial Z_\alpha(v_\alpha)} &= \sum_{v \notin V_\alpha} M(w) \left(\left(X(w) - \hat{X}(w) \right) \prod_{\alpha' \neq \alpha} Z_{\alpha'}(v_{\alpha'}) \right) \\ &= \Delta_\alpha(M \circ X) - \Delta_\alpha(M \circ \hat{X}) = 0 \end{aligned} \tag{15}$$

The solution of this equation leads to two related but different iterative schemata: *Multiplicative Update Rules* (MUR) and *Alternating Least Squares* (ALS). First, when we use gradient ascent we obtain MUR for $PLTF_{EU}$ similar to [7] as

$$Z_\alpha \leftarrow Z_\alpha \circ \Delta_\alpha(M \circ X) / \Delta_\alpha(M \circ \hat{X}) \tag{16}$$

Second, we can solve it directly that allows us to obtain ALS for $PLTF_{EU}$

$$\Delta_\alpha(M \circ X) = \Delta_\alpha(M \circ \hat{X}) \tag{17}$$

Note that $\hat{X}$ depends on Z_α. If there is no missing data ($M(w) = 1$ for all w), the result is available in closed form.

1.6 Matricization

Any element wise equation can be converted into the matrix form. We call this operation as 'matricization'. Note that *Matricization* is originally defined in [5, 6] as the operation of converting a multiway array into a matrix by reordering the column fibers. In this paper we refer to this definition as 'unfolding'. For matricization we use Einstein's summation convention where repeated indices are added over with some adaptations such as 'faster index last' convention for unfolding operations and definition for Khati-Rao product.

Example 3 (Derivation of matrix form update rules for the TUCKER3 decomposition) *We compute first the prediction in matrix form*

$$\hat{X}^{i,j,k} = \sum_{pqr} G^{p,q,r} A^{i,p} B^{j,q} C^{k,r} \tag{18}$$

$$(\hat{X}_{(1)})_i^{kj} = (G_{(1)})_p^{rq} A_i^p B_j^q C_k^r = \left((A G_{(1)})((C \otimes B)^T) \right)_i^{kj} \tag{19}$$

$$\hat{X}_{(1)} = A G_{(1)} (C \otimes B)^T \tag{20}$$

Now, Δ_{Z_α} for all α can also be represented in matrix form. The functions Δ_A and Δ_G are

$$\Delta_A(X) \equiv (X_{(1)})_i^{kj} B_j^q C_k^r G_p^{rq} \equiv X_{(1)}(C \otimes B) G_{(1)}^T \tag{21}$$

$$\Delta_G(X) \equiv (X_{(1)})_i^{kj} A_i^p B_j^q C_k^r \equiv A^T X_{(1)}(C \otimes B) \tag{22}$$

- *if KL ((14)) we evaluate $\Delta_\alpha(Q)$ and $\Delta_\alpha(M)$ where $Q = M \circ (X/\hat{X})$*

$$A \leftarrow A \circ \frac{Q_{(1)}(C \otimes B)G_{(1)}^T}{M_{(1)}(C \otimes B)G_{(1)}^T} \qquad G_{(1)} \leftarrow G_{(1)} \circ \frac{(A^T Q_{(1)})(C \otimes B)}{(A^T M_{(1)})(C \otimes B)} \tag{23}$$

- *if EUC-ALS. We solve $\Delta_\alpha(X) = \Delta_\alpha(\hat{X})$ (17) when there are no missing observations, i.e., $M(w) = 1$ for all w. We show only the updates for the core tensor G. The pseudo-inverse of A is denoted by $A^\dagger$. From (22) we have*

$$A^T X_{(1)}(C \otimes B) = A^T \left(AG_{(1)}(C \otimes B)^T\right)(C \otimes B) \tag{24}$$

$$G_{(1)} \leftarrow A^\dagger X_{(1)} \left((C \otimes B)^T\right)^\dagger \tag{25}$$

2 Model Selection for Tensors

We developed an model selection framework for $PLTF_{KL}$, that is for the non negative tensor factorization models although it can be extended for other error measures as well. We note that selecting the right generative model among many alternatives can be a difficult task. For example, given the observation $X^{i,j,k}$ one can propose CP generative model as $\hat{X}^{i,j,k} = \sum_r Z_1^{i,r} Z_2^{j,r} Z_3^{k,r}$, or a TUCKER3 model $\hat{X}^{i,j,k} = \sum_{p,q,r} Z_1^{i,p} Z_2^{j,q} Z_3^{k,r} Z_4^{p,q,r}$ or some arbitrary model as $\hat{X}^{i,j,k} = \sum_{p,q} Z_1^{i,p} Z_2^{j,p} Z_3^{k,q} Z_4^{p,q}$. Our model selection framework can be user both model order determination and model selection. Our selection method is based on getting a model score by lower bounding the log marginal likelihood by variational Bayes. The Bayesian approach offers an elegant solution based on computing marginal likelihood $p(X|\Theta)$, where latent variables and the parameters are integrated out as the exact computation is intractable [4, 2].

2.1 Variational Methods for $PLTF_{KL}$

Variational fixed point update equation for $Z_\alpha(v_\alpha)$ is, then, as follows

$$\langle Z_\alpha(v_\alpha)\rangle = \frac{A_\alpha(v_\alpha) + L_\alpha(v_\alpha) \sum_{v \notin V_\alpha} M(w) \frac{X(w)}{\hat{X}_L(w)} \prod_{\alpha' \neq \alpha} L_{\alpha'}(v_{\alpha'}))}{\frac{A_\alpha(v_\alpha)}{B_\alpha(v_\alpha)} + \sum_{v \notin V_\alpha} M(w) \prod_{\alpha' \neq \alpha} E_{\alpha'}(v_{\alpha'})} \tag{26}$$

where

$$E_\alpha(v_\alpha) = \langle Z_\alpha(v_\alpha)\rangle = C_\alpha(v_\alpha) D_\alpha(v_\alpha) \tag{27}$$

$$L_\alpha(v_\alpha) = \exp(\langle \log Z_\alpha(v_\alpha)\rangle) = \exp(\psi(C_\alpha(v_\alpha))) D_\alpha(v_\alpha) \qquad \psi \text{ is the digamma fn} \tag{28}$$

$$\hat{X}_E(w) = \sum_{\bar{w} \in \bar{W}} \prod_\alpha E_\alpha(v_\alpha) \tag{29}$$

$$\hat{X}_L(w) = \sum_{\bar{w} \in \bar{W}} \prod_\alpha L_\alpha(v_\alpha) \tag{30}$$

We come up with the following variational lower bound for the log marginal likelihood of the $PLTF_{KL}$ models as

$$\mathcal{L}(X|\Theta) \geq \mathcal{B} = \sum_{w \in W} -\hat{X}_E(w) - \log\Gamma(X(w)+1) + X(w) \log \hat{X}_E(w) \tag{31}$$

$$- KL[\mathcal{G}(C,D)||\mathcal{G}(A,B/A)] \tag{32}$$

where KL term can become $\sum_\alpha \sum_{v_\alpha \in V_\alpha} \log(C_\alpha(v_\alpha))$ asymptotically making theVB bound approaches to BIC score [1].

References

[1] M. J. Beal. *Variational Algorithms for Approximate Bayesian Inference*. PhD thesis, Gatsby Computational Neuroscience Unit, University College London, 2003.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[3] A. T. Cemgil. Bayesian inference for nonnegative matrix factorisation models. *Computational Intelligence and Neuroscience*, 2009:1–17, 2009.

[4] Z. Ghahramani and M. Beal. Propagation algorithms for variational bayesian learning. *In Neural Information Processing Systems*, 13, 2000.

[5] H. A. L. Kiers. Towards a standardized notation and terminology in multiway analysis. *Journal of Chemometrics*, 14:105–122, 2000.

[6] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[7] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. 13:556–562, 2001.

[8] Ruslan Salakhutdinov and Andriy Mnih. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, volume 20, 2008.

[9] M. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*,, 1:1–305, 2008.

[10] Y. Kenan Yılmaz and A. Taylan Cemgil. Probabilistic latent tensor factorization. In Vincent Vigneron, Vicente Zarzoso, Eric Moreau, Rémi Gribonval, and Emmanuel Vincent, editors, *Latent Variable Analysis and Signal Separation*, volume 6365 of *Lecture Notes in Computer Science*, pages 346–353. Springer, 2010.